Persistence of Perfect Numbers

Parker Emmerson

October 2024

1 Introduction

For any non-constant polynomial $P(n)$ with integer coefficients, where $P(n)$ outputs positive integers for all positive integers n, there exists an infinite number of integers n such that $P(n)$ is a perfect number.

Key Points:

- **Polynomial** $P(n)$: This is any polynomial function with integer coefficients, like $P(n) = n^2 + 3n + 2$, or any higher degree polynomial, as long as it does not reduce to a constant when considering its behavior over positive integers.

- **Perfect Numbers**: A number that is equal to the sum of its proper divisors (divisors excluding itself). The smallest examples are 6, 28, 496, and 8128.

- **Infinite Occurrences**: The conjecture claims that no matter the polynomial, you can always find infinitely many n where $P(n)$ evaluates to a perfect number.

Why This Conjecture is Significant:

- **Number Theory**:

It touches on deep questions in number theory regarding the distribution of special numbers like perfect numbers among the outputs of polynomial functions, which generally grow in complex ways.

- **Open Problem**: The distribution of perfect numbers is not well understood, and their appearance in sequences defined by polynomials could provide new insights or methods to study their properties.

- **Difficulty**: Proving or disproving such a conjecture would require understanding not just of how often perfect numbers occur naturally, but how often they can occur within the structured yet potentially erratic sequence generated by polynomials.

The Conjecture: For any non-constant polynomial $P(n)$ with integer coefficients, where $P(n)$ outputs positive integers for all positive integers n, there exists an infinite number of integers n such that $P(n)$ is a perfect number.

2 Sieve

5. Modified Sieve Construction:

- **Step 1:** Start with A.

- **Step 2:** For each small prime q, remove numbers from A that are not potentially of Euler's form or that don't satisfy some relaxed version of the perfect number condition (like having an almost correct divisor sum, considering computational limits).

- **Step 3:** Use bounds or heuristics on how often perfect numbers might appear. Given their sparsity, one might use probabilistic or heuristic arguments to estimate how many numbers to expect in A.

6. Complexity and Practicality: - **Efficiency**: Calculating $\sigma(n)$ for each n in A is computationally expensive for large N. - **Adaptation**: Traditional sieves are designed for properties like primality, where divisibility by primes directly reduces the set. Perfectness involves a sum of divisors, making direct sieving less straightforward.

Practical Implementation: - **Heuristics**: Use known distributions or patterns in perfect numbers to narrow down the search space within A. - **Hybrid Approach**: Combine computational checks for small n with theoretical

3 Polynomial

```
```python
import sympy as sp

def lagrange_interpolation(points):
    x = sp.Symbol('x')
    n = len(points)
    polynomial = sp.sympify(0)
    for i in range(n):
        xi, yi = points[i]
        basis = 1
        for j in range(n):
            if i != j:
                xj, _ = points[j]
                basis *= (x - xj) / (xi - xj)
        polynomial += yi * basis
    return polynomial

# Points corresponding to the first few perfect numbers
perfect_points = [(1, 6), (2, 28), (3, 496), (4, 8128), (5, 33550336)]

# Generate the polynomial
```
```

```
P = lagrange_interpolation(perfect_points)

print("The polynomial passing through the given perfect numbers:")
print(sp.expand(P))
```

The polynomial passing through the given perfect numbers:

$$\frac{16760347}{12}x^4 - \frac{83795017}{6}x^3 + \frac{586534205}{12}x^2 - \frac{418938659}{6}x + 33514406$$